



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/562,046	04/11/2006	Eckhard Kruse	034193-042	4285
21839 7590 01/11/2008 BUCHANAN, INGERSOLL & ROONEY PC POST OFFICE BOX 1404 ALEXANDRIA, VA 22313-1404				
EXAMINER NICKERSON, JEFFREY L.				
ART UNIT 4117		PAPER NUMBER		
NOTIFICATION DATE 01/11/2008		DELIVERY MODE ELECTRONIC		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ADIPFDD@bipc.com
debra.hawkins@bipc.com

Office Action Summary

Application No.

10/562,046

Applicant(s)

KRUSE ET AL.

Examiner

JEFFREY NICKERSON

Art Unit

4117

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 11 April 2006.
2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-20 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) 1-20 is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
10) ☒ The drawing(s) filed on 23 December 2005 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☒ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☒ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-8508)
Paper No(s)/Mail Date 23 December 2005
4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____
5) ☐ Notice of Inventor's Patent Application
6) ☐ Other: _____

DETAILED ACTION

1. This communication is in response to Application No. 10/562,046 filed nationally on 11 April 2006 and internationally on 10 April 2004. Claims 1-18 have been examined.

Specification

2. Applicant is reminded of the proper language and format for an abstract of the disclosure. The abstract should be in narrative form and generally limited to a single paragraph on a separate sheet within the range of 50 to 150 words. It is important that the abstract not exceed 150 words in length since the space provided for the abstract on the computer tape used by the printer is limited. The form and legal phraseology often used in patent claims, such as "means" and "said," should be avoided. The abstract should describe the disclosure sufficiently to assist readers in deciding whether there is a need for consulting the full patent text for details. The language should be clear and concise and should not repeat information given in the title. It should avoid using phrases which can be implied, such as, "The disclosure concerns," "The disclosure defined by this invention," "The disclosure describes," etc.

3. The abstract of the disclosure is objected to because it contains both implied and legal phraseology. The phrase "The invention relates to" in the first line falls under the category of implied phraseology. The phrases "means of a communication connection"

in lines 2-3, lines 4-5, and line 7 fall under the category of legal phraseology. Correction is required. See MPEP § 608.01(b).

4. The incorporation of essential material in the specification by reference to an unpublished U.S. application, foreign application, or patent is improper. Applicant is required to amend the disclosure to include the material incorporated by reference, **if the material is relied upon to overcome** any objection, rejection, or other requirement imposed by the Office. The amendment must be accompanied by a statement executed by the applicant, or a practitioner representing the applicant, stating that the material being inserted is the material previously incorporated by reference and that the amendment contains no new matter. In this case, the applicant is attempting to incorporate by reference the German application No. 103 32 360.0 filed on July 17, 2003. This fails to meet the requirement that non-English prior-filed applications must be accompanied with an English language translation. This is only being noted and no objection is made.

5. The disclosure is objected to because of the following informalities: incorrect spelling or grammar. Page 11, line 5 of the applicant submitted specification contains the phrase "communicate link", which should be changed to --communication link--.

Page 15, line 32 of the applicant submitted specification contains the phrase "evens", which should be changed to --events--. Appropriate correction is required.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

7. Claims 1-3, 5, 6, 9-14, 16, 17, and 19 are rejected under 35 U.S.C. 102(b) as being anticipated by Barker et al (US 6,363,421 B2).

Regarding claim 1, Barker teaches a method for managing and transmitting events from a server (Barker: Figure 3, item 32 depicts the element management system server) via a communication link (Barker: Figure 3, depicts communication using Http over tcp/ip or CORBA) to at least one client (Barker: Figure 3, item 28 depicts the element management system client) where: (Barker: abstract; See also Figures 2 and 3)

possible events are logged in a client event service (Java applets in client) for the purpose of initializing or updating the client, (Barker: col 4, lines 19-36 specifies applets using a CORBA interface to communicate to the EMS server in order to update web browser information regarding network elements)

possible events are logged in a server event service (various EMS server subcomponents, including, but not limited to, the Object server and the Alarm manager) for the purpose of initializing or updating the server, (Barker: See Figure 4; col 4, lines

37-55 specify the EMS server handles the majority of networking management including trapping and real time notification management)

detected events which have been logged are transferred from an installation interface (SNMP API or mediator) to the server event server (Barker: See Figure 4, item Trap Daemon; col 9, line 23 - col 10, line 50 specifies how HP Openview plays a role as the SNMP manager to detect events),

requests initiated by the client event service regarding the detected events are made to the server event service, (Barker: col 11, lines 21-28 specify clients register with the Event Distributor to receive filtered events)

on the basis of a request which has been made to the server event service the detected events are transmitted to the client event service, (Barker: col 11, lines 21-28 specify the Event Distributor distributes events to clients based on their specified filters)

events received by the client event service are transmitted to a client application (Barker: See Figure 2, Java applets pass information back to web browser for display; See also Figure 15; col 4, lines 18-36 specify the web browsers are running java applets to obtain/manage the information passed back)

Regarding claim 2, Barker teaches wherein the events to be transmitted are detected by a data capture unit (SNMP agent) in a technical installation (network element) and are reported to the installation interface of the server. (Barker: col 17, lines 60-65; col 19, lines 13-23; col 19, line 55 – col 20, line 5; See also Figure 4, item 14 subcomponents)

Regarding claim 3, Barker teaches wherein the client applications logs a client callback function in the client event service for every event about which it is to be notified, (Barker: Figure 6, term Client Callback Function definition) and the client event service uses the communication link to log (initialize) a SNMP trap. (Barker: col 25, lines 40 – col 26, line 10 specifies that the EMS server contains an attribute table for network elements it monitors. When initializing the table immediately after a client registration, the EMS polls the traps via the SNMP mediator. The SNMP mediator delivers the data back to the EMS attribute manager with callback functions)

Regarding claim 5, Barker teaches wherein after a client callback function has been logged for the first time the client logging function starts a request generator which then makes requests for event transmission to the server event service. (Barker: col 11, lines 21-28 specifies the clients register for event notifications with the Event Distributor at the EMS server; col 22, lines 25-43 specify this registration for notification is done by passing a callback object reference, which means the client has inherently created a callback function with which the reference points to)

Regarding claim 6, Barker teaches wherein the request generator of the client event service makes the requests for event transmission to the server event service cyclically. (Barker: col 19, lines 39-54 specify the object server is capable of handling periodic polling from clients)

Regarding claim 9, this system claim comprises limitations substantially similar to that of claim 1 and the same rationale of rejection is used, where applicable.

Regarding claim 10, this system claim comprises limitations substantially similar to that of claim 2 and the same rationale of rejection is used, where applicable.

Regarding claim 11, Barker teaches wherein the server event service has at least one server callback function which can be logged for at least one event and which is called when an event for which it is logged occurs. (Barker: col 25, lines 40 – col 26, line 10 specifies that once the client indicates which events should be watched, the initialization of the attribute table is done by server callback functions with the traps via the SNMP mediator)

Regarding claim 12, Barker teaches wherein the server event service has at least one server logging function for logging server callback functions (Barker: col 25, lines 40 – col 26, line 10 specifies that the EMS server contains an attribute table for network elements it monitors. When initializing the table immediately after a client registration, the EMS polls the traps via the SNMP mediator. The SNMP mediator delivers the data back to the EMS attribute manager with callback functions), at least one server event table for holding data records which describe a respective logging operation, (Barker: col 21, line 63 - col 22, line 23 specify the storage of service objects, which contain attribute information about network elements) and at least one event queue for holding

entries which describe a respective event. (Barker: col 10, line 53-67 specifies the use of event queues)

Regarding claim 13, Barker teaches wherein the server event service has, for every client event service with which it communicates via a communication link, a separate client data record which respectively contains at least one server event table (Barker: col 17, lines 27-50 specifies that when clients register, a record is created that contains their Session and Application ID, Callback object, and Filter object, which is an event table) and at least one event queue. (Barker: col 10, line 53-67 specifies the use of event queues)

Regarding claim 14, Barker teaches wherein the server event service has a tidying function which deletes the client data record if the associated client event service is no longer communicating with the server event service. (Barker: col 16, lines 62-67 specify removal if determined to be inactive)

Regarding claim 16, Barker teaches wherein the client has at least one client callback function which can be logged for at least one event and which is called when the event for which it is logged occurs. (Barker: col 22, lines 25-43 specify using a client callback function to pass notifications back when an event occurs)

Art Unit: 4117

Regarding claim 17, Barker teaches wherein the client event service has at least one client logging function for logging client callback functions, (Barker: col 25, lines 12-40 specify that a client registers for monitoring attributes/events by providing the managed object instance identifier, its attribute codes, and the callback function for delivery of changes; This means that a logging function must inherently be called from the GUI to log the user choices; See Figures 10-13 and col 7, lines 37-67), at least one client event table for holding data records which describe the log, (Barker: Figure 12, see table), and at least one request generator for making cyclic requests for event transmission. (Barker: col 19, lines 39-54 specify the object server is capable of handling periodic polling from clients)

Regarding claim 19, this method claim comprises limitations substantially similar to that of claim 3 and the same rationale of rejection is used, where applicable.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Art Unit: 4117

9. Claims 4, 7, 8, 15, 18, and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Barker et al (US 6,363,421 B2) as applied to the claims above, and further in view of Panikatt et al (US 6,349,333 B1) and Kampe et al (US 2002/0016867 A1) .

Regarding claim 4, Barker teaches wherein to log the callback functions for an event with which the same event name is associated with the client and with the server in preparation for the method, the following steps are performed:

the client applications calls a client logging function from the client event service and provides said function with the name of the event in question and with a pointer to the client callback function which is to be logged (Barker: col 25, lines 12-40 specify that a client registers for monitoring attributes/events by providing the managed object instance identifier, its attribute codes, and the callback function for delivery of changes; This means that a function must inherently be called from the GUI to log the user choices; See Figures 10-13 and col 7, lines 37-67),

the client logging function generates a unique event identifier (filter) and transmits the event identifier and the event name (attributes) via the communication link to a server logging function of the server event service, (Barker: col 11, lines 21-29 specify that the event distributor receives the event filter from the client; col 7, lines 57-63 specify registration for attributes changes via the filter; See also col 17, lines 25-50)

the server logging function logs a server callback function with the installation interface by transferring the event (attribute) name (Barker: col 25, line 60 – col 26, line

10 specifies that when a client first registers, the server must register the SNMP mediator and inform it which attributes need to be polled via a callback function),

the server logging function stores a data record which contains at least the event identifier and a pointer to the client callback function which is to be logged, in a server event table, (Barker: col 17, lines 27-50 specifies that when clients register, a record is created that contains a filter to identify events and a callback pointer; See also Figure 6, Callback Object Reference term)

the server logging function reports the performance of the logging operation to the client logging function of the client event service via the communication link, (Barker: col 33, lines 42-50 specify the use of TRAP acknowledgements; col 7, lines 56-63 specify these are subsequently passed back to the client as notifications)

the client logging function logs the client callback function by storing a data record in a client event table (Barker: Figure 12, see table), the data record containing at least the event identifier (Barker: Figure 12, see table, contains IDs and filter type).

Though Barker does teach the use of server callback functions for internal communications, Barker's invention differs slightly from the instant application. Instead of logging a client callback pointer in the client event table and logging a server callback pointer in the server event table, Barker passes the client callback pointer to the server and stores the client callback pointer in the server event table.

Panikatt, in a similar field of endeavor, teaches wherein the server event table contains a server callback function. (Panikatt: col 15, lines 23-45 specifies logging callbacks in the Management Information Server for callbacks to the JMA server, which

Art Unit: 4117

together operate as the alarm notification server; See Figure 6, item 613 and col 7, lines 13-25)

Panikatt does not teach wherein the client event table maintains client callback pointers.

Kampe, in a similar field of endeavor, teaches wherein the client event table (subscriber node's ES) contains a client callback pointer. (Kampe: [0047] and [0054] specify the subscriber node registers a callback function in preparation for receiving events on specific event channels; See Figures 1 and 2 for clarification on what Kampe refers to as an "Event Server")

It would have been obvious to one of ordinary skill in the art at the time the invention was made to utilize the teachings of Panikatt for registering a server callback function pointer in the server event table and the teachings of Kampe for registering a client callback function pointer in the client event table. The teachings of Panikatt and Kampe, when implemented in the Barker system, will allow one of ordinary skill in the art to protect function calls to be maintained locally. One of ordinary skill in the art would be motivated to utilize the teachings of Panikatt and Kampe in the Barker system in order to provide for a more secure system by requiring callbacks to be called locally, rather than allowing a remote server across the network perform the callback.

Regarding claim 7, the Barker/Panikatt/Kampe method teaches wherein events are transmitted by performing the following steps:

the installation interface detects an event which has occurred and calls the server callback function logged for this event, (Barker: col 25, line 60 – col 26, line 10),

the server callback function produces an entry describing the event in at least one event queue (Panikatt: col 7, lines 55-65 and Figure 6 specify an RMI alarm log skeleton and event dispatcher to forward the events to the client; Barker: col 10, line 53-67 specifies the use of event queues),

upon the next request from the client event server for event transmission the server event service reads the entry produced from the event queue and transmits it via the communication link to the client event service (Panikatt: col 7, line 66 – col 8, line 7 specifies the client requests the event information and then it is retrieved),

the client event service takes the entry received and ascertains and calls the client callback function logged for this event (Kampe: [0047] – [0048] specifies obtaining an event, passing it through a filter, then calling the corresponding callback function),

the client callback function executes a defined action for the corresponding event in the client application. (Kampe: [0048] specifies the callbacks are for throttling; [0054] specifies the callbacks pass the event on to the appropriate application; See also Figure 2)

Regarding claim 8, this system claim comprises limitations substantially similar to that of claim 14 and the same rationale of rejection is used, where applicable.

Art Unit: 4117

Regarding claim 15, this system claim comprises a subset of limitations substantially similar to that of claim 4 and the same rationale of rejection is used, where applicable, and wherein the event table is in the form of a hash table. (Panikatt: col 9, lines 39-50 specifies specifying using a hash table for storage of records)

Regarding claim 18, this system claim comprises a subset of limitations substantially similar to that of claims 4 and 15 and the same rationale of rejection is used, where applicable.

Regarding claim 20, this method claim comprises limitations substantially similar to that of claim 7 and the same rationale of rejection is used, where applicable.

Cited Pertinent Prior Art

10. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- a. Allavarpu et al (US 7,010,586 B1) discloses a system and method for event subscriptions using a CORBA gateway and a MIS.
- b. Angal et al (US 6,298,378 B1) discloses an event distribution system using an MIS.
- c. Brinnand et al (US 6,430,616 B1) discloses a system for scaling MIS command and response queues.

- d. Cook (US 5,621,892) discloses a method and apparatus for managing network events and distribution.
- e. Galloway et al (US 6,378,004 B1) discloses a method for asynchronous communication of device events.
- f. Kekic et al (US 6,788,315 B1) discloses a platform for managing a network and event trapping and distribution.
- g. McGuire et al (US 7,107,497 B2) discloses a method and system for event publication and subscription through the use of queues.
- h. Musante et al (US 7,152,104 B2) discloses a method and apparatus for event notification in a distributed computer system.
- i. Roytman et al (US 6,356,282 B2) discloses an alarm manager for a distributed network environment.
- j. Sondur et al (US 6,243,746 B1) discloses a method for using network topology objects for event distribution.
- k. Weisman et al (US 2002/0112058 A1) discloses a system for discovering connected devices and using events to manage registration of device information.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to JEFFREY NICKERSON whose telephone number is (571)270-3631. The examiner can normally be reached on M-Th, 8:30-6:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Beatriz Prieto can be reached on 571-272-3902. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/J.N./
Jeffrey Nickerson
Patent Examiner

/Prieto, Beatriz/
Supervisory Patent Examiner, Art Unit 4117